

Implementasi *Encoder* dan *Decoder Hamming* pada DSK TMS320C6416T

Anggy Kusuma Dewi Wisma, Suwadi, Titiek Suryani

Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111

E-mail: suwadi110@ee.its.ac.id, titiks@ee.its.ac.id

Abstrak—Untuk menghasilkan suatu sistem komunikasi yang handal, dalam artian bebas dari *error*, perlu diterapkan suatu algoritma kode yang dapat mendeteksi kesalahan (*error detection*) sekaligus memperbaiki kesalahan bit (*error correction*), dengan menggunakan kode *Hamming* kesalahan (*error*) dapat di koreksi dan di deteksi dengan mudah, hal ini bertujuan untuk meningkatkan kehandalan dan untuk memperoleh suatu cara penyelesaian yang optimal. Kode *Hamming* mampu untuk mengoreksi semua kesalahan tunggal dalam satu blok. Pada Tugas akhir ini dilakukan implementasi *Encoder* dan *Decoder Hamming* pada TMS320C6416 yang merupakan salah satu modul dasar untuk pengembangan dan evaluasi aplikasi sistem komunikasi digital waktu nyata buatan *Texas Instruments*. Pemrograman menggunakan Simulink yang terdapat dalam *software* MATLAB. Analisis meliputi nilai BER yang diperoleh dengan mengubah parameter Eb/No dari kesalahan per blok (pola kesalahan) melalui kanal ideal (AWGN) dengan menggunakan modulasi BPSK. Dari hasil Analisis yang dilakukan bahwa probabilitas error kanal sebelum dilakukan pengkodean *Hamming* dapat diperkecil setelah dilakukan pengkodean *Hamming*. Kode *Hamming* (7,4) menunjukkan kinerja lebih baik dibanding kode *Hamming* lainnya terbukti pada EbNo 6 Kode *hamming* (7,4) sudah mencapai BER 0.

Kata Kunci—*Hamming*, *Encoder*, *Decoder*, *error detection*, TMS320C6416

I. PENDAHULUAN

Pada pengiriman data, kanal transmisi dapat menimbulkan kesalahan (*error*) yang merupakan masalah pada sistem komunikasi, sebab dapat mengurangi kinerja dari sistem. Salah satu upaya yang dilakukan adalah dengan menerapkan *error control coding*. Keuntungan dari adanya *error control coding* adalah tidak diperlukannya pengiriman kembali data dalam suatu pengiriman. *Error correcting code* juga digunakan dalam pengamanan data dalam jumlah besar untuk menghindari kerusakan pada data yang bersangkutan. Kode *Hamming* merupakan salah satu jenis *error correcting code* yang sederhana, kode *Hamming* banyak digunakan pada berbagai peralatan elektronik.

Kode *Hamming* ini diimplementasikan TMS320C6416. Alasan pemilihan TMS320C6416 dari tipe-tipe yang lain karena kapasitas memori yang lebih tinggi, kecepatan pemrosesan data dan memiliki beberapa perangkat tambahan didalamnya. Didalam TMS320C6416 dibangun *encoder* dan *decoder*. Kemudian *encoder* dan *decoder* ini dievaluasi dengan

mengubah-ubah sinyal yang diterima oleh penerima, artinya sama dengan berubahnya Eb/No, kode ini mampu mengurangi kesalahan. *encoder* dan *decoder* *Hamming* disimulasikan dengan menggunakan *simulink* dengan bantuan *software* MATLAB

II. TEORI PENUNJANG

A. Kode Hamming

Teknik pengkodean *Hamming* memiliki keunggulan dimana dapat tepat mengoreksi satu kesalahan bit yang timbul, Kode ini memiliki *block code* (n,k) dengan parameter :

$$\begin{aligned} \text{Panjang kode (jumlah bit blok codeword): } n &= 2^m - 1 \\ \text{Jumlah bit informasi : } k &= 2^m - m - 1 \end{aligned}$$

$$\begin{aligned} \text{Jumlah bit pariti : } n - k &= m \\ \text{kapasitas untuk koreksi error : } t &= 1 \quad (d_{\min} = 3) \\ d_{\min} &= \text{Jarak kode minimum} \end{aligned}$$

1) Encoding Hamming

Bit *steam* dari sumber data yang masuk ke *encoder* dikodekan dengan menggunakan suatu generator. langkah pertama yang harus dilakukan adalah menentukan (n) dan (k). Nilai (n) dan (k) diperoleh dengan rumus $(n, k) = (2^m - 1, 2^m - m - 1)$. Setelah diketahui nilai polinomialnya, lalu susun matriks generator nya dengan tahapan sebagai berikut:

- Gunakan polinomial sebagai baris pertama dan tambahkan nilai 0 hingga jumlah bit seluruhnya menjadi sebanyak n. Pola bit yang dihasilkan diberi nama dengan $g(x)$.
- Buat baris kedua dengan menggeser bit terakhir pada baris pertama ke bit pertama, prosesnya adalah $x \cdot g(x)$.

Buat baris ketiga dengan menggeser bit terakhir pada baris kedua ke bit pertama, prosesnya adalah $x^2 \cdot g(x)$. Pergeseran bit terakhir terus dilanjutkan sampai sejumlah k baris. Matriks G yang terbentuk adalah sebagai berikut:

$$G = \begin{bmatrix} g(x) \\ x \cdot g(x) \\ x^2 \cdot g(x) \\ x^3 \cdot g(x) \end{bmatrix}$$

Generator yang terbentuk merupakan generator yang tidak sistematis. Untuk mengubahnya menjadi sistematis yaitu dengan operasi baris elementer. Matriks dari G yang sistematis merupakan generator dari kode *Hamming*.

$$G = \begin{bmatrix} g(x) \\ x \cdot g(x) \\ x^2 \cdot g(x) + g(x) \\ x^3 \cdot g(x) + g(x) + x \cdot g(x) \end{bmatrix}$$

Encoder kode Hamming akan menghasilkan n bit *output codeword* yang terdiri dari k bit informasi dan m bit *parity*. Pada proses *encoding* kode Hamming, n bit *output codeword* dihasilkan dengan mengalikan k bit matriks informasi dengan generator matriks yang mempunyai ukuran matriks $n \times k$. Proses tersebut dapat didefinisikan dengan persamaan berikut:

$$C = [M][G] \quad (2.1)$$

dimana : C = *Output codeword*;
 M = Matriks bit informasi;
 G = Matriks generator kode blok.

2) Decoding Hamming

Fungsi dari sebuah *dekoder* kode *Hamming* adalah untuk mendeteksi kesalahan *syndrome error* dan juga untuk memperbaiki kesalahan yang timbul pada *syndrome error* tersebut. Untuk melakukan koreksi kesalahan pada kode *Hamming*, harus diketahui matriks *parity check* nya yang dinyatakan dengan persamaan :

$$H = [I | P^T] \quad (2.2)$$

dimana : H = matriks cek *parity*
 I = matriks identitas
 P^T = hasil transpose dari matriks *parity* P s

Dimana matriks *parity* cek diatas dapat dihitung sindrom dengan persamaan : $S = e \cdot HT$ (2.3)

dimana : S = Bentuk *syndrome error*
 e = Bentuk *error code*
 HT = Matriks *transpose parity check*.

Untuk mengetahui *error* dalam *codeword*, dilakukan dengan membandingkan *syndrome error* dari kode masing-masing jenis *Hamming* dengan *syndrome error codeword* yang diterima. *Syndrome error* diperoleh dengan mengalikan *codeword* yang diterima dengan transpose matriks cek *parity*.

3) Deteksi Error^[5].

Nilai *syndrome error* menentukan ada tidaknya *error* pada *codeword* yang diterima. Bila *syndrome error* bernilai nol, maka kode yang diterima tidak terkena *error*. Sehingga, kode yang dikirimkan (r) = kode yang diterima (e). Sedangkan bila *syndrome error* tidak bernilai nol, maka ada *error* terdeteksi. Lakukan pencarian posisi *error* berdasarkan posisi ke berapa pola bit *syndrome* tersebut pada kolom matriks *transpose parity check* (HT). Langkah berikutnya adalah mengoreksi bit-bit yang terkena *error*, yakni dengan menginvert nilai dari kode yang diterima, bila 0 menjadi 1, dan sebaliknya, sehingga diperoleh kode yang dikirimkan. Untuk mendapatkan kembali bit-bit informasi dalam *codeword*, dilakukan pembuangan

semua bit-bit *parity* nya^[6]. Misalnya sebuah generator *Hamming* dengan *parity* (m) = 3

Maka : $n = 2m - 1 = 7$

$$k = 2m - m - 1 = 4$$

$$(n, k) = (2m - 1, 2m - m - 1) = (7, 4)$$

Sehingga merupakan kode Hamming (7,4). Berdasarkan Tabel 2.1, maka diperoleh pola bit Polinomial GF(2) = 1 1 0 1

$$g(x) = 1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0$$

$$x \cdot g(x) = 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0$$

$$x^2 \cdot g(x) = 0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0$$

$$x^3 \cdot g(x) = 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1$$

Diperoleh generator matriks G yang tidak sistematis :

$$G = \begin{bmatrix} g(x) \\ x \cdot g(x) \\ x^2 \cdot g(x) \\ x^3 \cdot g(x) \end{bmatrix} = \begin{bmatrix} 1101000 \\ 0110100 \\ 0011010 \\ 0001101 \end{bmatrix}$$

Dengan operasi baris elementer, diperoleh generator matriks G yang sistematis :

$$G = \begin{bmatrix} g(x) \\ x \cdot g(x) \\ x^2 \cdot g(x) + g(x) \\ x^3 \cdot g(x) + g(x) + x \cdot g(x) \end{bmatrix} = \begin{bmatrix} 1101000 \\ 0110100 \\ 1110010 \\ 1010001 \end{bmatrix}$$

Misalnya data yang diberikan oleh sumber data adalah $M = 1 \ 0 \ 0 \ 0$. Maka, bit *output codeword* yang dihasilkan oleh *encoder* kode *Hamming*: $C = [M] \cdot [G]$ (2.4)

$$= [1000] \begin{bmatrix} 1101000 \\ 0110100 \\ 1110010 \\ 1010001 \end{bmatrix} \\ = [1101000]$$

Suatu kode Hamming (7,4) memiliki matriks *parity check* :

$$H = \begin{bmatrix} 1001011 \\ 0101110 \\ 0010111 \end{bmatrix}$$

Bila kode yang diterima (e) = 1 1 0 1 0 0 0, maka bentuk *syndrome error* dari kode yang diterima: $S = e \cdot HT$

$$= [1101000] \begin{bmatrix} 100 \\ 010 \\ 001 \\ 110 \\ 011 \\ 111 \\ 101 \end{bmatrix} \\ = [000]$$

Karena *syndrome error* bernilai nol, maka kode yang diterima tersebut tidak terkena *error*. Sehingga, kode yang dikirimkan (r) = kode yang diterima (e) = 1 1 0 1 0 0 0. Setelah dilakukan pembuangan semua bit-bit *parity* nya, diperoleh kembali bit-bit informasi dalam *codeword* / data yang dikirim

M = 1 0 0 0

4) Modulasi dan Demodulasi BPSK

Modulasi PSK (*Phase Shift Keying*) merupakan modulasi digital dengan mengubah fase sinyal pembawa (*carrier*) sesuai dengan perubahan kondisi masukan dari sinyal informasi [1].

Proses demodulasi dapat dibagi menjadi tiga subbagian besar. Pertama, masukan yang berupa sinyal gelombang ditekan oleh sinyal pembawa secara alami, sehingga diperlukan deteksi koheren. Pembawa fase koheren berasal dari sinyal yang masuk ini disebut *carrier recovery*. Dalam pengiriman bit yang dimodulasi, dibagian penerima bisa terjadi kesalahan yang dapat dinyatakan dalam probabilitas error P_e . Untuk modulasi BPSK ini, probabilitas error-nya dinyatakan dalam rumus berikut.

$$P_e = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \quad (2.5)$$

Dimana nilai E_b/N_0 bisa dicari melalui rumus berikut.

$$\frac{E_b}{N_0} = \frac{S}{N R} \quad (2.6)$$

S merupakan daya sinyal, sedangkan N adalah daya noise yang terukur. Untuk variabel W dan R masing-masing adalah *bandwidth* dan laju bit yang digunakan.

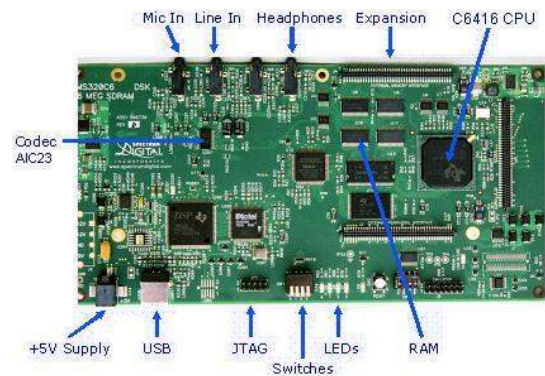
B. AWGN (*Additive White Gaussian Noise*)

AWGN (*Additive White Gaussian Noise*) merupakan suatu proses stokastik yang terjadi pada kanal dengan karakteristik memiliki rapat daya spectral noise merata di sepanjang range frekuensi. AWGN mempunyai karakteristik respon frekuensi yang sama disepanjang frekuensi dan variannya sama dengan satu. Pada kanal transmisi selalu terdapat penambahan derau yang timbul karena akumulasi derau termal dari perangkat pemancar, kanal transmisi, dan perangkat penerima. Derau yang menyertai sinyal pada sisi penerima dapat didekati dengan model matematis statistik AWGN. Derau AWGN merupakan gangguan yang bersifat Additive atau ditambahkan terhadap sinyal transmisi, dimodelkan dalam pola distribusi acak Gaussian dengan mean (m) = 0, standar deviasi (σ) = 1, power spectral density (pdf) = $N_0/2$ (W/Hz), dan mempunyai rapat spektral daya yang tersebar merata pada lebar pita frekuensi tak berhingga.

C. BER

BER (*Bit Error Rate*) adalah jumlah kesalahan bit dibagi dengan jumlah bit yang ditransfer selama interval waktu tertentu. BER merupakan ukuran performansi unitless atau tidak mempunyai ukuran, sering dinyatakan dalam prosentase.

BER dapat dianalisa dengan menggunakan simulasi komputer stokastik. Jika saluran transmisi Model dan sumber data model sederhana diasumsikan, BER juga dapat dihitung secara analitis. Dalam saluran bernoise, BER sering dinyatakan sebagai fungsi dari normalisasi rasio ukuran *carrier- to-noise* dilambangkan E_b/N_0 atau E_s/N_0 . Sebagai contoh, dalam kasus modulasi BPSK dan kanal AWGN, BER sebagai fungsi E_b/N_0 diberikan oleh :



Gambar 1. DSK TMS320C6416T

$$BER \approx Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \quad (2.11)$$

dimana : E_b : energy bit
 N_0 : kerapatan noise

D. DSK TMS320C6416T (*DSP Starter Kit*)

DSK TMS320C6416T [5] merupakan multi-layer board berukuran 8.75 x 4.5 inch (210 x 115 mm) yang disuplai daya eksternal +5 volt. [3] DSK ini terdiri dari sebuah 1 GHz DSK TMS320C6416T fixedpoint digital signal processor dan 16 bit stereo codec TLV 320AIC23 untuk input dan output. Codec AIC23 menyediakan ADC dan DAC dengan clock 12 MHz dan sampling rate 8 – 96 kHz. DSK meliputi 16 MB pada Synchronous Dynamic Random (SDRAM) dan 256 kB pada flash memory. Terdapat empat konektor pada DSK untuk input dan output :

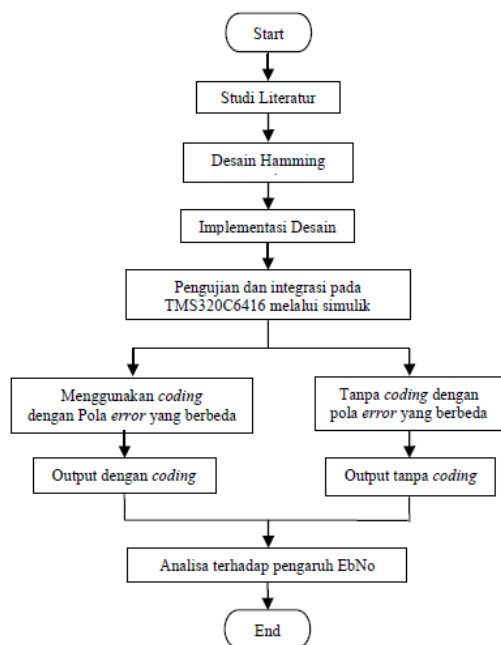
- MIC IN untuk input dari microphone.
- LINE IN untuk input dari function generator,
- LINE OUT untuk output , dan
- HEADPHONE untuk output pada headphone.

Dip switch dalam DSK dapat difungsikan sesuai program dan menyediakan fungsi kontrol. Terdapat pengatur tegangan yang menyediakan 1,26 V untuk processor dan 3,3 V untuk sistem memori dan kelengkapan lain.

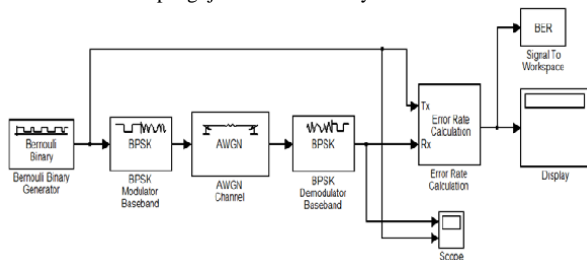
TMS320C6416 pada DSK adalah floating point processor yang didasari pada arsitektur VLIW (Very-Long-Instruction-Word). Memori internal meliputi 2 level penyimpanan daya, yang terdiri dari 32 KB pada level penyimpanan program (L1P) dan level 1 penyimpanan data (L1D), dan 1024 kB pada level 2 pembagian memori antara program dan data. Selain itu, juga mempunyai sebuah interface untuk memerintah synchronous dan memori (SDRAM dan SBSRAM) dan asynchronous memori (SPRAM dan EPROM). Memiliki 8 fungsi atau execution units dalam 6 ALU dan 2 multiplier units, 32 - bit address bus ke 4 GB (gigabytes) address, dan 2 set 32 - bit general - purpose registers.

III PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada bab ini menjelaskan tentang gambaran sistem yang akan dibuat dan diuji dimana secara umum meliputi perangkat lunak, perangkat keras, simulasi sistem, proses implementasi



Gambar 2. Flow chart pengujian dan analisis system



Gambar 3 Blok diagram simulasi modulasi BPSK

sistem, dan metode pengujian sistem. Berikut skema metodologi yang digunakan dalam tugas akhir ini.

A Pemodelan Sistem

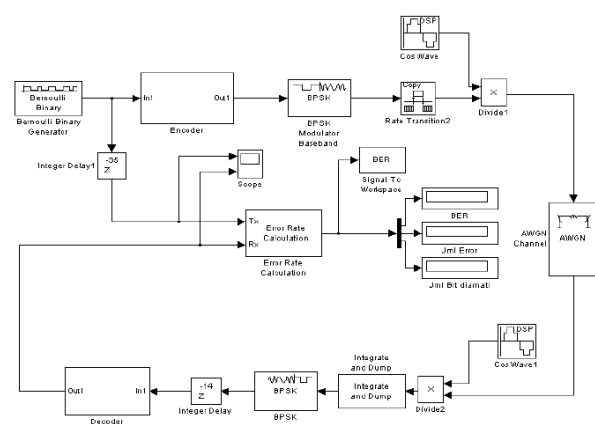
Sebelum melakukan implementasi sistem ke DSK TMS, terlebih dahulu dimodelkan pada simulink matlab. Sistem yang akan digunakan terlihat pada gambar 10 berupa rangkaian yang digunakan untuk pengujian. Pemodelan sistem ini bertujuan untuk memperoleh gambaran sistem yang akan disimulasikan dan juga diimplementasikan.

1) Simulasi modulasi BPSK tanpa menggunakan pengkodean Hamming

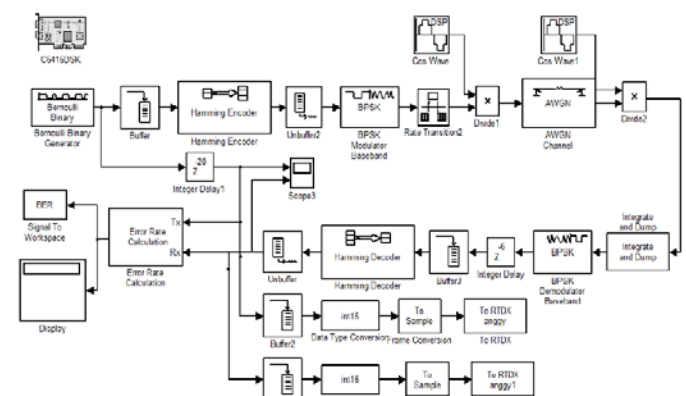
Simulasi modulasi BPSK dilakukan Sebagai acuan dalam melakukan simulasi ini, kesalahan bit terjadi karena *noise* yang terdapat pada kanal AWGN. Pada blok AWGN *Channel* terdapat beberapa parameter yang harus diisi dengan teliti. Seperti contohnya pada *Symbol Period*, nilai harus sama dengan periode bit yang berasal dari sumber. BER diukur dengan cara mengubah variasi nilai Eb/N0 pada kanal AWGN.

2) *Simulasi kode Hamming menggunakan kanal AWGN dan Modulasi BPSK Passband*

Blok diagram simulasi untuk modulasi *passband* dirancang untuk mendekati implementasi secara riil. Hal ini dapat



Gambar 4. Blok Diagram Simulasi kode *Hamming* dengan modulasi BPSK passband dan kanal AWGN



Gambar 5. Blok implementasi kode *Hamming*

dilakukan dengan mengalikan sinyal keluaran hasil modulasi dengan sinyal sinus atau *sin wave*. Pada bagian penerima juga dilakukan pengalihan sinyal, hal ini bertujuan untuk mendapatkan sinyal asli hasil modulasi.

B. Implementasi Sistem Kode Hamming

Implementasi sistem kode *Hamming* ini menggunakan sumber sinyal yang dibangkitkan dari *Simulink* dan menggunakan kanal AWGN. Seperti yang telah dirancang pada simulasi sistem, sumber sinyal menggunakan blok *Bernoulli Binary Generator*. Performasi sistem tetap dilakukan dengan mengevaluasi nilai BER.

Penambahan kanal AWGN bertujuan untuk melihat performansi sistem kode *Hamming* yang telah diimplementasikan pada DSK TMS320C6416. Performansi sistem akan dilakukan dengan cara mengubah nilai E_b/N_0 pada blok kanal AWGN sehingga didapatkan *Bit Error Rate* dari sistem kode *Hamming*. Berikut blok sistem yang akan diimplementasikan pada DSK TMS320C6416.

Pada implementasi ini menggunakan 2 perangkat yaitu Komputer (PC) dan DSK TMS320C6416. PC digunakan untuk memprogram file simulink dan melihat hasil keluaran sistem yang telah diimplementasikan. Gambar 3.27 merupakan skema pengujian pada implementasi yang dilakukan .

C. Implementasi sistem kode Hamming pada TMS320C6416T

Dengan memperhatikan penjelasan-penjelasan di atas sudah dapat menggambarkan cara implementasi sistem kode *Hamming* pada DSK TMS320C6416T. Disini akan lebih dijelaskan lebih rinci dan terstruktur mengenai urutan langkah-langkah implementasi hingga persiapan pengujiannya. Adapun langkah-langkahnya antara lain:

- Siapkan blok diagram simulink matlab yang akan diintegrasikan ke DSK board.
- Pastikan PC sudah terhubung dengan perangkat DSK TMS320C6416T.
- Buka software *6416 DSK CCStudio v3.3*.
- Lakukan proses integrasi dengan cara mengklik menu 'Incremental build' pada simulink.

D. Metode Pengujian Sistem dan Pengolahan Data

Semua sistem akan diuji pada laju bit 1 kbps dan dengan kondisi jumlah bit yang dikirimkan adalah sebanyak 1.000.000 bit. Sistem ini akan dilewatkan kedalam kanal AWGN yang divariasikan nilai E_b/N_0 . Variasi nilai E_b/N_0 dari 0 dB sampai 10 dB. Apabila terdapat kesalahan pada sisi penerima, maka jumlah bit salah akan dibandingkan dengan jumlah bit yang dikirim sehingga nantinya didapatkan nilai BER (Bit Error Rate). Metode pengujian ini dilakukan dengan menggunakan sistem RTDX dimana sistem ini akan diletakkan pada sisi pemancar dan sisi penerima seperti yang telah dijelaskan pada bab sebelumnya. Nilai yang telah didapat dari RTDX ini akan diolah melalui Matlab sehingga didapatkan nilai BER dengan nilai variasi E_b/N_0 yang diubah-ubah.

IV PENGUJIAN DAN ANALISIS SISTEM

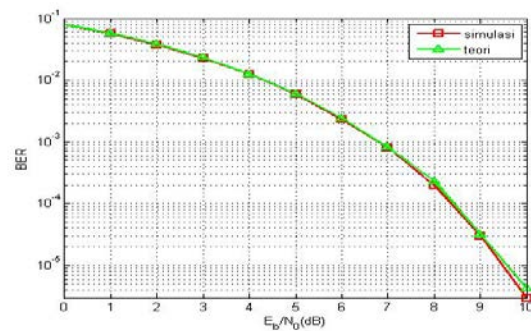
A. Kinerja Modulasi BPSK secara Teori dan Simulasi

Perhitungan dilakukan menggunakan rumus BER atau *Probability Error* modulasi BPSK menggunakan tabel Q-function. Penggunaan hasil modulasi BPSK tanpa pengkodean ini akan menjadi acuan dalam membandingkan kinerja sistem kode *Hamming*. Hasil kinerja sistem dengan modulasi BPSK tanpa menggunakan pengkodean adalah seperti ditunjukkan pada gambar 4.1.

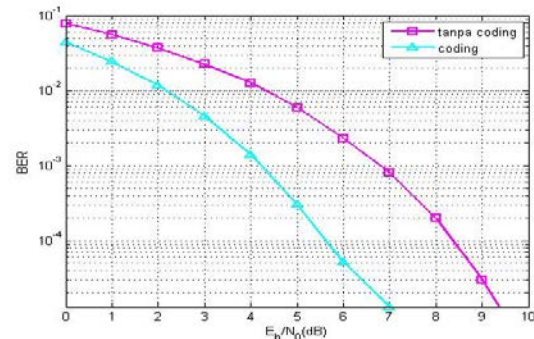
Pada grafik 4.1 dapat dilihat bahwa tidak ada perbedaan antara perhitungan BER modulasi BPSK secara teori maupun simulasi. Ini membuktikan bahwa simulasi modulasi BPSK menggunakan Matlab Simulink telah sesuai dengan teorinya. Untuk hasil yang lebih jelas dapat dilihat pada tabel di bawah ini. Pada simulasi yang dilakukan jumlah bit yang dikirimkan adalah 1,000,000 bit.

B. Pengujian kode Hamming dengan tanpa kode Hamming

Simulasi yang Berikut perbandingan nilai BER dengan atau tanpa menggunakan kode *Hamming* dengan jumlah bit yang dikirim sebesar 1.000.000 bit. Pada simulasi ini kesalahan bit terjadi karena adanya noise pada kanal AWGN dengan merubah-ubah nilai E_b/N_0 akan didapatkan pengaruhnya pada nilai BER. Gambar 4.3 merupakan perbandingan nilai BER



Gambar 6. Grafik perbandingan BER modulasi BPSK secara teori dan simulasi



Gambar 8. Grafik perbandingan BER dengan menggunakan *Hamming* dan tanpa *Hamming*

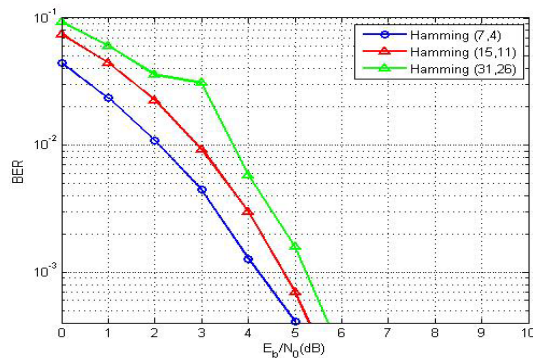
antara BPSK menggunakan kode *hamming* dengan tidak menggunakan kode *hamming*.

Dari grafik diatas dapat dilihat bahwa nilai BER pada kode *Hamming* lebih kecil dari sistem tanpa pengkodean ini membuktikan bahwa dengan diberi kode *Hamming* hasil yang diperoleh jauh lebih baik dibandingkan tanpa kode *Hamming* dalam mendeteksi dan memperbaiki *error* yang terjadi. Berdasarkan nilai pada grafik pada saat BER 10^{-3} coding gain nya adalah 2,8 dB.

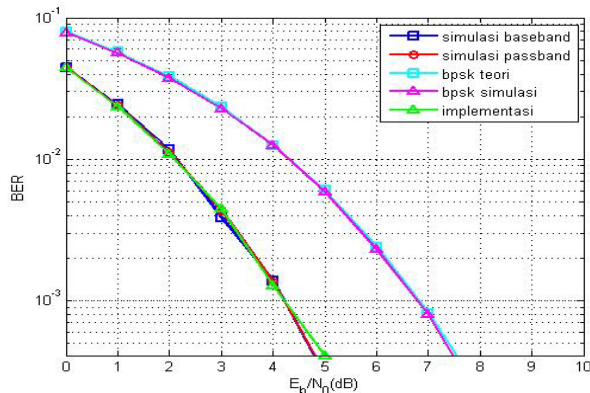
C. Pengujian Sistem Implementasi

Pada implementasi ini ditambahkan blok RTDX untuk mendapatkan nilai BER dalam pengujian, Blok ini ditempatkan pada sisi pengirim dan penerima. Jumlah bit yang dikirim adalah sebanyak 100.000 bit, berbeda dengan pada waktu simulasi yakni 1.000.000 bit. Hal ini disebabkan alat yang digunakan yakni TMS320C6416 tidak memungkinkan untuk mencapai jumlah bit yang sama.

Dari data hasil sistem implementasi kode *Hamming* diatas dapat disimpulkan bahwa semakin bertambah nilai perubahan *Energy of bit signal power Noise* (E_b/N_0) yang terjadi pada sistem pengkodean tersebut, maka semakin mempengaruhi nilai kesalahan bit informasi yang diterima pada sisi penerima dimana diukur oleh Blok *Error Rate Calculation*. BER yang terhitung akan semakin mendekati nol, untuk nilai pengukuran BER yang semakin kecil dan mendekati nol berarti kesalahan dalam penerimaan bit informasi akan semakin mendekati harga kebenaran.



Gambar 9. Grafik Perbandingan implementasi sistem kode *Hamming* pada TMS320C6416



Gambar 10. Grafik BER kode *Hamming*

D. Analisa Sistem Keseluruhan

Setelah dilakukan pengujian pada simulasi dan implementasi sistem kode *Hamming*. Untuk implementasi pengujian dilakukan berasal dari *simulink* dengan masukan *bernoulli binary generator* dengan laju 1 bit kbps, BER hasil implementasi hampir mendekati nilai BER pada Matlab. Sistem keseluruhan dapat dianalisis dengan membandingkan grafik BER vs E_b/N_0 pada kode *Hamming* (7,4).

Pengujian BER pada implementasi lebih baik dibandingkan simulasi baik *baseband* maupun *passband*, disebabkan adanya filter pada TMS320C6416 serta jumlah bit yang diamati berbeda-beda. Jumlah bit yang dikirimkan pada tahap pengujian implementasi adalah 100.000 bit, sedangkan pada tahap simulasi jumlah bit yang dikirimkan sebanyak 1000.0000 bit. Kinerja yang terbaik terdapat pada kode *Hamming* (7,4) yaitu pada kondisi E_b/N_0 6 dB didapatkan BER bernilai 0.

V. KESIMPULAN

A. Kesimpulan

Dari hasil perancangan, simulasi, dan implementasi yang telah dilakukan pada Tugas akhir ini dapat diambil kesimpulan :

1. Pada tugas akhir ini telah berhasil diimplementasikan kode *Hamming* (7,4), *Hamming* (15,11), dan *Hamming* (31,26) dan menghitung nilai BER (*bit error rate*) dengan memvariasikan nilai E_b/N_0 pada kanal AWGN.
2. Kode *hamming* hanya mampu memperbaiki kesalahan bit (*error correction*) terhadap satu buah error (*single error correcting*) saja.

3. Nilai BER pada kode *Hamming* lebih kecil dari sistem tanpa pengkodean sehingga pada saat BER 10^{-3} coding gain nya didapatkan 2,8 dB.
4. Kinerja dari pengkodean *Hamming* mengalami peningkatan seiring dengan penambahan bit paritas.
5. Kinerja kode *Hamming* terbaik terdapat pada implementasi *Hamming* (7,4) terbukti bahwa pada E_b/N_0 6 dB sudah mencapai BER 0.

DAFTAR PUSTAKA

- [1] Sklar, Bernard. "Digital Communications Fundamentals and applications". Prentice Hall. California. 2001.
- [2] Wicker, S.B., *Error Correcting Codes for Digital Communications and Storages*, Prentice Hall, 1998.
- [3] Lin, S., D. J. Costello, *Error Control Coding: Fundamentals and Applications*, Prentice Hall, 1983
- [4] Oppenstein, Michael. 2008. *Code Hamming (7,4) discussion and Implementation*. Prentice hall: New York
- [5] Simatupang, Efendi. "Analisis BER Pada Sistem Modulasi BPSK Untuk Peningkatan Kinerja Kanal AWGN Dengan Kode Hamming". Medan, Maret 2010.
- [6] Panjaitan, Sihar P. "Teori Informasi dan Pengkodean". Buku Kedua. Halaman 1-22. Medan, 2008
- [7] Chassaing, Rulph. "Digital Signal Processing and Applications with the TMS320C6713 and TMS320C6416 DSK". JOHN WILEY & SONS, INC. USA. Second Edition, 2008.
- [8] ———. "TMS320C6416T DSK Technical Reference". SPECTRUM DIGITAL, INC. 2004.
- [9] Hayes, Monson H. *Schaum's Outline of Theory and Problems of Digital Signal Processing*. McGraw Hill, United State of America. 1999.